

```
#####
#
# http://oeis.org/A002487
#
a <- function(n) {
  b <- as.numeric(intToBits(n))
  l <- sum(b)
  m <- which(b == 1) - 1
  d <- 1
  if(l > 1) for(j in 1:(l-1)) d[j] <- m[j+1] - m[j] + 1
  f <- c(0, 1)
  if(l > 1) for(j in 3:(l+1)) f[j] <- d[j-2] * f[j-1] - f[j-2]
  return(f[l+1])
}
#
# intToBits(n), n < 2^32
#
#####
#
# Alternative to intToBits(n): binary(n) due to
# M.J.CRAWLEY, 2007. The R Book, John Wiley & Sons, Ltd, England
#
binary <- function(n) {
  i <- 0
  string <- numeric(64)
  while(n > 0) {
    string[64 - i] <- n %% 2
    n <- n %/% 2
    i <- i + 1
  }
  first <- match(1, string)
  string[first:64]
}
#
#####
#
# http://oeis.org/A002487
#
a <- function(n) {
  b <- rev(binary(n))
  B <- sum(b)
  m <- which(b == 1) - 1
  c <- 1
  if(B > 1) for(j in 1:(B-1)) c[j] <- m[j+1] - m[j] + 1
  f <- c(0, 1)
  if(B > 1) for(j in 3:(B+1)) f[j] <- c[j-2] * f[j-1] - f[j-2]
  return(f[B+1])
}
#
# Example
n <- 73
a(n)
#
A002487 <- NULL
for(n in 1:63) A002487 <- c(A002487, a(n))
A002487
#
#####
```

```
#####
#
# http://oeis.org/A007306
#
# Given n, compute directly a(n)
# by taking into account the binary representation of n-1
#
#
aa <- function(n) {
  b <- rev(binary(n))
  B <- sum(b)
  m <- which(b == 1) - 1
  c <- 1
  if(B > 1) for(j in 1:(B-1)) c[j] <- m[j+1]-m[j]+1
  f <- c(1, m[1]+2) # c(0, 1) in A002487
  if(B > 1) for(j in 3:(B+1)) f[j] <- c[j-2]*f[j-1]-f[j-2]
  return(f[B+1])
}
#
# Example
n <- 73
aa(n-1)
#
A007306 <- NULL
for(n in 1:63) A007306 <- c(A007306, aa(n))
A007306
#
#####
```